

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra kybernetiky a biomedicínského inženýrství

**Testovací systém pro před-servisní
detekci poruchy v mobilním telefonu**

**Testing System for Pre-service Error
Detection in Cell Phones**

Zadání bakalářské práce

Student: **Matěj Kaloč**

Studijní program: B2649 Elektrotechnika

Studijní obor: 2612R041 Řídicí a informační systémy

Téma: **Testovací systém pro před-servisní detekci poruchy v mobilním telefonu**
Testing System for Pre-service Error Detection in Cell Phones

Jazyk vypracování: čeština

Zásady pro vypracování:

1. Rozbor problematiky hardwarových komponent mobilního přístroje a programových sekvencí pro jejich verifikaci.
2. Návrh struktury testovacího systému pro rozpoznání poruch jednotlivých komponent.
3. Realizace testovacího systému pro před-servisní detekci poruchy v mobilním telefonu.
4. Srovnání měřených a analyzovaných výsledků s teoretickými předpoklady.
5. Zhodnocení dosažených výsledků.

Seznam doporučené odborné literatury:

- [1] *Learn android studio: build android apps quickly and effectively*. Berkeley: Apress, 2014. ISBN 978-1430266013.
- [2] MEDNIEKS, Zigurd R. *Programming Android*. 2nd ed. Beijing: O'Reilly, c2012, xvii, 542 s. ISBN 978-1449316648.
- [3] VÁVRŮ, Jiří a Miroslav UJBÁNYAI. *Programujeme pro Android*. 2., rozš. vyd. Praha: Grada, 2013, 250 s. Průvodce (Grada). ISBN 978-80-247-4863-4.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Zdeněk Macháček, Ph.D.**

Datum zadání: 01.09.2015

Datum odevzdání: 29.04.2016

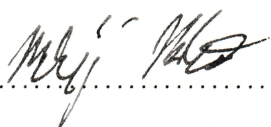


doc. Ing. Jiří Koziorek, Ph.D.
vedoucí katedry

prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 29. dubna 2016


.....

Z důvodů utajení firemního tajemství společnosti PEGATRON Czech s.r.o.
jsou kapitoly popisující návrh, realizaci a verifikaci vyvinutého systému označeny jako neveřejné, dále
jsou i všechny přílohy neveřejné.
Celý text této práce je uchován na Katedře kybernetiky a biomedicínského inženýrství.

PEGATRON ©
PEGATRON Czech s.r.o.
Na Rovince 862, 720 00 Ostrava-Hrabovské
Ludomír Neusser IČ: 26730847 DIČ: CZ26730847

Rád bych na tomto místě poděkoval všem, kteří mě při psaní práce podporovali. Zvláště pak vedoucímu práce Ing. Zdeňku Macháčkovi, Ph.D. za dobrou komunikaci a zapůjčení laboratoře.

Abstrakt

Tato práce řeší problematiku testování hardwarových komponent mobilních telefonů s operačním systémem Android. Program v mobilním telefonu či tabletu projde s uživatelem jednotlivé komponenty a otestuje je buď automaticky nebo s pomocí uživatele. Výsledek testů poté odešle do databáze. Dle výsledku testů operátor rozhodne, zda-li je vhodné mobilní telefon odeslat na reklamaci, nebo jestli vše funguje, jak má. Díky této aplikaci se zefektivní reklamace mobilních telefonů a tabletů.

Klíčová slova: Android, detekce poruchy, aplikace, reklamace

Abstract

This bachelor thesis deals with testing hardware components of cell phones with operating system Android. The program in cell phone or tablet goes through components of phone with user and tests them automatically or with the aid of user. Test's results are sent to database. Operator decides whether it is suitable to send cell phone to reclamation or it is ok according to test results. This application makes the reclamation of cell phones more efficient.

Key Words: Android, error detection, application, reclamation

Obsah

Seznam použitých zkratek a symbolů	8
Seznam obrázků	9
Seznam tabulek	10
1 Úvod	11
2 Teoretický rozbor operačního systému Android, hardwarových komponent telefonů a požadavků na program PegaTest	12
2.1 Popis architektury OS Android	12
2.2 Programování aplikací pro OS Android	14
2.3 Výčet cílů tvorby aplikace PegaTest	14
2.4 Rozbor hardwarových komponent telefonů a jejich softwarového vybavení	19
3 Rozbor aplikace	23
4 Verifikace dosažených výsledků	24
5 Závěr	25
Literatura	26

Seznam použitých zkratk a symbolů

OS	– Operační Systém
ART	– Android Runtime
SDK	– Software Development Kit
SDT	– Android Development Tools
SIM	– Subscriber Identification Module
NFC	– Near Field Communication
GPS	– Global Positioning System
RAM	– Random-access Memory
CPU	– Centrální Procesorová Jednotka
USB	– Universal Serial Bus
MAC	– Media Access Control
PRN	– Pseudo Random Number
PHP	– Hypertext Preprocessor
HTTP	– Hypertext Transfer Protocol

Seznam obrázků

1	Architektura OS Android [2]	12
---	-----------------------------	----

Seznam tabulek

1	Obecné parametry	19
2	Parametry displeje	19
3	Rozměry	20
4	Fotoaparát	20
5	Baterie	20
6	Procesor a paměť	20
7	Obecné parametry	21
8	Parametry displeje	21
9	Rozměry	21
10	Fotoaparát	21
11	Procesor, paměť a baterie	22

1 Úvod

Se stále větším zastoupením zařízení s OS Android na našem trhu je stále větší potřeba tato zařízení opravovat, nastane-li nějaká porucha. Každá firma prodávající takové zařízení se tak snaží být v něčem lepší než její konkurenti. Jedna z výhod, které firma může svým zákazníkům nabídnout, je rychlejší servis v případě poruchy. Jednou z částí servisu je lokalizace poruch v zařízení a právě o to se stará aplikace, která je předmětem mé bakalářské práce.

Téma jsem si vybral proto, že jsem se už dlouhou dobu chtěl naučit programovat aplikace pro platformu Android. Proto když se naskytla příležitost vytvářet plnohodnotnou aplikaci, bral jsem to jako skvělou příležitost. Dalším důvodem bylo to, že jsem chtěl, aby můj výtvar měl nějaký smysl a byl používán i po obhajobě mé práce.

Cílem práce je vytvořit plnohodnotnou aplikaci, která usnadní čas mnoha lidem. Technikům usnadní hledání problému, který mají opravit, a uživateli objasní, jestli je chyba na jeho straně, nebo jestli je telefon vadný.

Aplikace je pojmenovaná PegaTest a vyvíjel jsem ji pro firmu Pegatron od dubna 2015. Firma Pegatron se zabývá servisem Set-top boxů a nově i servisem mobilních telefonů ZenFone od firmy Asus. Aplikace má být vytvořena ve dvou verzích. První verze pro zákazníka. Tato verze má být pomalejší, testy musí být více vysvětleny a výsledek odeslán do databáze. Druhá verze pro technika má být svižnější a nemá obsahovat prvky, které by test prováděný zkušeným technikem zbytečně zpomalovaly.

Idea verze aplikace pro zákazníka je taková, že zákazník před tím, než navštíví servis, zavolá operátorovi. Operátor zákazníkovi nabídne nějakou výhodu při reklamaci výrobku, jako je například rychlejší průběh reklamace za to, že si zákazník nainstaluje aplikaci PegaTest, která ověří, zda periferie, o které se zákazník domnívá, že je vadná, skutečně nefunguje. Aplikace ověří i ostatní hardware telefonu a výsledek zašle do databáze firmy. Operátor poté posoudí, jestli je reklamace vhodná nebo jestli je problém na straně klienta. V případě, že dojde k reklamaci, může technik nahlédnout do databáze a zjistit tak, co všechno je vadné a je třeba opravit.

Verze aplikace pro technika má sloužit k systematické kontrole jednotlivých periférií mobilního telefonu nebo k ověření funkčnosti jen daných periférií.

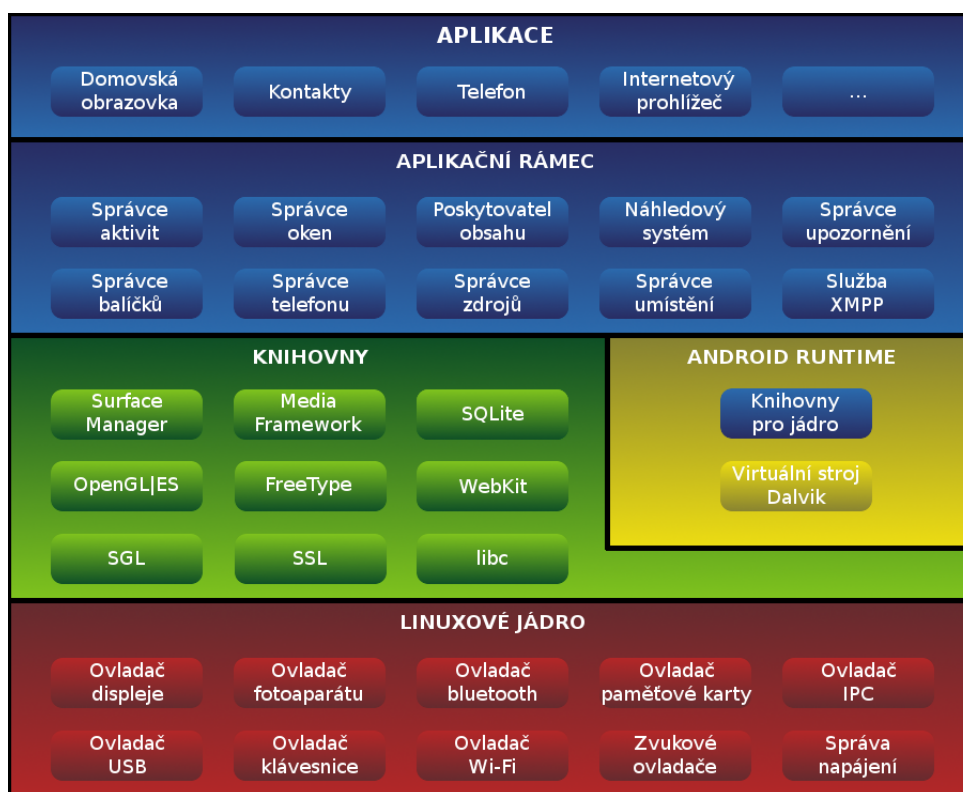
První kapitola nazvaná teoretický rozbor operačního systému Android, hardwarových komponent telefonů a požadavků na program PegaTest se zabývá rozбором operačního systému Android, na jehož platformě program vznikl a také rozбором hardwarových komponent mobilů, pro něž byla aplikace vyvinuta. V druhé kapitole, rozbor aplikace, se nachází návrh struktury programu a realizace struktury této aplikace. Ve třetí části práce jsou zhodnoceny výsledky testování této aplikace a tyto jsou porovnány s teoretickými předpoklady.

2 Teoretický rozbor operačního systému Android, hardwarových komponent telefonů a požadavků na program PegaTest

Android je mobilní operační systém založený na jádře Linuxu, který je dostupný jako otevřený software. Při vývoji systému byla brána v úvahu omezení, kterými disponují klasická mobilní zařízení, jako výdrž baterie, menší výkonnost a málo dostupné paměti. Zároveň bylo jádro Androidu navrženo pro běh na různém hardwaru. [1]

2.1 Popis architektury OS Android

Architektura operačního systému Android je rozdělena do pěti vrstev. Každá vrstva pracuje samostatně a má svůj účel. V praxi však nemusí být jednotlivé vrstvy přímo odděleny od ostatních a dochází mezi nimi ke spolupráci. [1, 11]



Obrázek 1: Architektura OS Android [2]

2.1.1 Popis jádra systému

Jádro je základem operačního systému a především zajišťuje komunikaci mezi hardwarem a softwarem. Mezi jeho hlavní součást patří Ovladače, které zařizují právě onu komunikaci. Při startu je jádro zavedeno do operační paměti a je mu předáno řízení. Může tak zajistit správu

procesů, správu paměti, správu napájení, síťové spojení atd. Jádro systému Android je postaveno na jádře systému Linux. V prvních verzích Androidu šlo o verze Linuxu 2.6.x, pozdější verze využívají novější linuxová jádra.[1, 11]

2.1.2 Rozbor vrstvy knihovny

Další vrstvou jsou knihovny, které jsou napsány v C nebo C++ kódu a využívají je různé komponenty systému. Tyto funkce jsou vývojářům poskytnuty prostřednictvím Android Application Framework. Výčet základních knihoven:

- OpenGL ES - knihovna sloužící k vykreslování 3D grafiky a akceleraci výpočtů
- FreeType - knihovna pro vykreslování bitmapového a vektorového písma
- libc - knihovna pro vestavěné zařízení
- SGL - knihovna sloužící k vykreslování 2D grafiky
- SQLite - relační databázová knihovna
- SSL - knihovna pro využití šifrovacího protokolu
- LibWebCore - knihovna webového prohlížeče [1, 11]

2.1.3 Popis Android Runtime

Android Runtime zajišťuje kompilaci instalované aplikace napsané v jazyce Java do nativních instrukcí. Pro kompilaci se používá dopředná kompilace, což je následník virtuálního stroje Dalvik, který nahradila ve verzi Android 5.0. Dopředná kompilace znamená, že se aplikace napsaná v Java kódu přeloží do Java byte kódu, poté do byte kódu Dalviku a při instalaci se zkompiluje do nativního kódu zařízení. Důvodem je úspora energie a zrychlení běhu aplikací. Nevýhodou je delší doba instalace. Dalvik zůstává nadále součástí ART z důvodu zpětné kompatibility. [1, 3, 4, 5]

2.1.4 Popis Application framework

Tato vrstva je pro vývojáře nejdůležitější. Poskytuje přístup k velkému počtu služeb, které mohou být použity přímo v aplikacích. Tyto služby mohou zpřístupňovat data v jiných aplikacích, prvky uživatelského rozhraní, upozorňovací stavový řádek, aplikace běžící na pozadí, hardware používaného zařízení a mnoho dalších služeb a funkcí. Základní sada služeb:

- View System - umožňuje použití prvků, jako jsou tlačítka a textová pole v uživatelském rozhraní
- Content Providers - umožňuje přístup k obsahu jiných aplikací, například poznámkám v kalendáři a kontaktům

- Resource manager - poskytuje přístup zdrojům, které nejsou součástí kódu například obrázkům, textovým řetězcům a grafickému rozhraní
- Notification manager - umožňuje aplikacím zobrazit upozornění ve stavovém řádku
- Activity manager - řídí životní cyklus aplikací, jejich spuštění, průběh a ukončení a také poskytuje orientaci v zásobníku s aplikacemi
- Package manager - obsahuje informace o aplikacích nahrených do operačního systému [1, 11]

2.1.5 Rozbor vrstvy Aplikace

Nejvyšší vrstvu tvoří aplikace, které využívají běžní uživatelé, ať už se jedná o aplikace předinstalované, stažené z Android Marketu nebo nainstalované jiným způsobem. Příkladem takové aplikace může být kalendář, webový prohlížeč nebo kalkulačka.[1]

2.2 Programování aplikací pro OS Android

Android software development kit je sada nástrojů pro programování Android aplikací. Tato sada obsahuje debugger, knihovny, emulátor, dokumentaci, ukázkový kód a tutoriály. Oficiálním SDK pro OS Android je Android Studio. Předchůdcem Android Studia je vývojové prostředí Eclipse s ADT pluginem. Aplikace se dají také programovat v jakémkoli textovém editoru a kompilovat pomocí kompilačních programů.[1, 6, 7]

2.2.1 Popis Eclipsu s ADT Pluginem

Eclipse je open source vývojová platforma. Flexibilní návrh této platformy dovoluje rozšířit seznam podporovaných programovacích jazyků za pomoci pluginů, například o C++ nebo PHP a pomocí ADT pluginu také o možnost programování Android aplikací. Eclipse s ADT pluginem byl oficiálním SDK pro android do konce roku 2014. Poté ho nahradilo Android Studio, nicméně mnoho vývojářů stále používá Eclipse s ADT Pluginem. [7, 8, 9, 10]

2.2.2 Popis Android Studia

Android Studio je nové vývojové prostředí založené na IntelliJ IDEA. Android studio bylo firmou Google oficiálně představeno 16. května 2013. Od června 2013 je zdarma k dispozici pro uživatele.[6, 7]

2.3 Výčet cílů tvorby aplikace PegaTest

Bylo specifikováno logické rozložení jednotlivých funkčních částí aplikace a dvacet pět jednotlivých testů.

2.3.1 Popis obecných požadavků

- Aplikace automaticky detekuje jazyk OS, a pokud je aplikace do tohoto jazyka přeložena, používá tuto jazykovou mutaci aplikace. Výchozím jazykem je angličtina.
- Po spuštění aplikace se zobrazí dvě tlačítka „PegaTest“ a „PegaTest - PRO“.

2.3.1.1 Tlačítko PegaTest Tlačítko spustí všechny testy v náhodném pořadí. Tato funkce bude pravděpodobně vše, co bude zpřístupněno běžnému uživateli.

2.3.1.2 Tlačítko PegaTest - PRO Po kliknutí na možnost „PegaTest - PRO“ aplikace ukáže nabídku s těmito tlačítky:

- „Zpět“ - Tlačítko pro návrat do hlavní nabídky.
- „Spustit test“ - Tato funkce spustí všechny testy v náhodném pořadí, stejně jako tlačítko „PegaTest“ v hlavní nabídce.
- „Výběr testu“ - Funkce zobrazí seznam všech testů a uživatel si může jeden z nich zvolit.
- „Pořadí testů“ - Po kliknutí na toto tlačítko se zobrazí seznam všech testů s možností zvolit si, které testy a v jakém pořadí se provedou.
- „Nastavení“ - Možnost úpravy některých parametrů jednotlivých testů.
- „Konec“ - Tato funkce má ukončit aplikaci.

2.3.2 Popis požadavků na testy

Firma Pegatron zaslala tyto požadavky testy:

2.3.2.1 Wi-Fi Tento test má za úkol ověřit funkčnost Wi-Fi přijímače a také otestovat rychlost sítě.

Po spuštění funkce se otevře okno se seznamem Wi-Fi sítí v dosahu. Pokud se uživateli v daném časové limitu nepodaří připojit k žádné Wi-Fi síti, test skončí. Poté, co se uživatel úspěšně připojí k Wi-Fi síti dialogové okno zmizí a začne se stahovat obrázek z internetu. Pokud se obrázek stáhne do daného časového limitu, zobrazí se na displeji, pod obrázkem se vypíše doba a průměrná rychlost stahování obrázku a aplikace se zeptá uživatele, zda se obrázek stáhl správně. Uživatel zvolí, jestli se obrázek stáhl správně, a tím test ukončí. Celý test jde v jakékoli fázi přeskočit tlačítkem „Skip“. Celkový čas testu je nanejvýš šedesát sekund, poté dojde k vypršení časového limitu a test končí.

2.3.2.2 Tlačítka Tento test otestuje funkčnost hardwarových tlačítek telefonu.

Po spuštění testu se na displeji zobrazí seznam hardwarových tlačítek s číslem, kolikrát má tlačítka uživatel zmáčknout. Po zmáčknutí tlačítka se číslo udávající počet zmáčknutí sníží o jedna. Pokud všechny ukazatele zmáčknutí ukazují nula, test se ukončí. Další možností ukončení testu je vypršení časového limitu, který je stanoven na třicet sekund.

2.3.2.3 Dotykový displej Test dotykového displeje si klade za cíl otestovat dotykový displej.

Po spuštění testu se na displeji zobrazí 10x20 bílých obdélníků s černými okraji. Po kliknutí na obdélník daný obdélník zčerná. Poté, co uživatel klikne na všechny obdélníky, test skončí. Další možností ukončení testu je vypršení časového limitu třicet sekund.

2.3.2.4 Zdířka na SIM kartu Tento test otestuje funkčnost zdířky na SIM kartu.

Po spuštění testu aplikace sama zkontroluje, zda je v zařízení vložená SIM karta. V případě, že ano, ukončí test, v případě, že ne, sdělí uživateli, že v zařízení není přítomna SIM karta a nechá uživatele zvolit, jestli test přeskočí, nebo ukončí s negativním výsledkem. Další možností ukončení testu je vypršení časového limitu 45 sekund.

2.3.2.5 Zadní kamera Test otestuje, zda zadní kamera fotí správně.

Po spuštění se na displeji zobrazí obraz ze zadní kamery a tlačítko na vyfocení. Po vyfocení se zobrazí vyfocený obraz a dotaz na uživatele, zda je fotografie bez viditelných chyb. Poté se vše ještě dvakrát zopakuje a výsledek testu je zaslán do databáze a testování se ukončí. Další možností ukončení testu je vypršení časového limitu šedesát sekund.

2.3.2.6 Přední kamera Test přední kamery testuje přední kameru stejným způsobem, jakým je testována zadní kamera, viz 2.3.2.5. Jediný rozdíl je v tom, že se na displej nepřenáší obraz zadní kamery, ale přední.

2.3.2.7 Kompas Tento test si klade za cíl otestovat vnitřní kompas zařízení.

Po spuštění se na displeji zobrazí obrázek kompasu, který ukazuje na sever. Při otáčení mobilním telefonem se kompas otáčí tak, aby stále ukazoval na sever a okolo kompasu se zobrazuje kruh. Kruh znázorňuje, kterým směrem vůči telefonu kompas již ukazoval. V případě, že je kruh kompletní, test je ukončen. Test se také ukončí v případě, že vyprší časový limit třicet sekund. Uživatel může test kdykoli přeskočit.

2.3.2.8 Akcelerometr Test má ověřit funkčnost vnitřního senzoru zrychlení.

Po spuštění aplikace vyzývá v náhodném pořadí uživatele k tomu, aby telefon postavil zpříma, naklopil na pravou a levou stranu, otočil spodní částí vzhůru, položil na stůl displejem nahoru a displejem dolů. V případě, že tak uživatel učiní, test se ukončí. V případě, že tak

neučiní, test se ukončí po vypršení časového limitu třiceti sekund nebo může být v průběhu testu přeskóčen uživatelem.

2.3.2.9 Gyroskop Tento test ověří správné fungování vnitřního gyroskopu v zařízení.

Aplikace nejprve vyzve uživatele, aby hýbal s telefonem, tím se ověří, že telefon dostává nějaké hodnoty ze senzoru. Poté je uživatel vyzván, aby telefon položil na stůl a potvrdí potvrzovacím tlačítkem. Po potvrzení je měřeno, zda senzor posílá do telefonu nějaké hodnoty. Pokud neposílá, test proběhl v pořádku, jestli posílá, je senzor špatný. Test může také ukončit časomíra nastavená na 45 sekund nebo uživatel tlačítkem „Skip“.

2.3.2.10 Sluchátko Tento test ověří funkčnost reproduktoru sluchátka.

Po spuštění testu se spustí hudba přehrávána přes reproduktor sluchátka. Každých pět sekund se navíc přehraje jedno ze čtyř slov. Poté, co uživatel uslyší slovo, klikne na displeji na tlačítko „Jsem připraven“ a poté vybere ze slov to správné. Při vybrání správného slova se vše ještě třikrát zopakuje, V případě, že uživatel vybere špatné slovo, celý test končí. Test může také ukončit časovač nastavený na čtyřicet pět sekund.

2.3.2.11 Reproduktor Test má ověřit funkčnost reproduktoru. Funguje obdobně jako test Sluchátka, viz 2.3.2.10, s tím rozdílem, že zvuk je přehráván reproduktorem, a ne sluchátkem telefonu.

2.3.2.12 Senzor přiblížení Test má ověřit funkčnost senzoru přiblížení, který je obvykle umístěn na přední části telefonu vedle sluchátka. Senzor slouží k tomu, aby při telefonování zamkl obrazovku a ta nemohla být ovládána uchem volajícího.

Aplikace nejdříve zjistí, zda je senzor přikrytý, jestli ano, napíše na displeji, ať ho uživatel odkryje a při odkrytí ukončí test. Jestli při startu senzor zakrytý není, aplikace napíše na displej, aby uživatel senzor zakryl a po zakrytí ukončí test. Test může skončit i kvůli vypršení časovače nastaveného na patnáct sekund.

2.3.2.13 Světelný senzor Světelný senzor slouží k získání hodnoty osvětlení okolí, a podle této hodnoty se nastavuje jas displeje telefonu. Tento test ověřuje jeho funkčnost.

Aplikace vyzve uživatele, aby zakryl senzor. Poté, co je senzor zakryt, vyzve uživatele, aby přiblížil telefon ke zdroji světla. Pokud senzor po dobu testu ukazuje správné hodnoty, test je ukončen. V případě, že neukazuje, test ukončí časomíra nastavená na patnáct sekund.

2.3.2.14 Blesk fotoaparátu Tento test ověřuje funkčnost blesku fotoaparátu v mobilním telefonu a je prováděn jen v případě, že telefon má blesk.

Aplikace začne rozsvěcet a zhasínat blesk v sekundových intervalech a na displeji se zeptá uživatele, jestli blesk funguje. Test se ukončí po uživatelské odpovědi, nebo ho ukončí časomíra nastavená na patnáct sekund.

2.3.2.15 Konektor na sluchátka Test ověřuje funkčnost konektoru na sluchátka.

Aplikace nejdříve vyzve uživatele k připojení sluchátek. Po připojení začne ve sluchátkách hrát hudba a test pokračuje stejně jako test telefonního sluchátka, viz 2.3.2.10. V případě, že jsou sluchátka odpojena, test se přeruší a aplikace vyzve uživatele k jejich opětovnému připojení.

2.3.2.16 USB konektor Test slouží k ověření funkčnosti USB konektoru.

Aplikace vyzve uživatele k propojení telefonu s počítačem pomocí USB kabelu. Po propojení počítače a telefonu vyzve uživatele k odpojení telefonu od počítače a vše opakuje ještě dvakrát. Potom, co je počítač třikrát detekován, test končí. Další možností ukončení testu je vypršení časomíry nastavené na třicet sekund.

2.3.2.17 Display Test souží k ověření funkčnosti displeje.

Aplikace zobrazí přes celou obrazovku barevné pruhy, šedé pruhy, bílou barvu, černou barvu a dva obrázky. Poté se zeptá uživatele, zda viděl nějaké poruchy obrazu nebo vadné pixely. Po uživatelské odpovědi test skončí.

2.3.2.18 Bluetooth Test má ověřit funkčnost bluetooth modulu v mobilním zařízení.

Aplikace vyzve uživatele, aby se pomocí bluetooth připojil k libovolnému zařízení. Po připojení se test ukončí. Test má časový limit čtyřicet pět sekund.

2.3.2.19 NFC Test otestuje, zda NFC senzor funguje správně.

Při spuštění testu je uživatel vyzván k aktivaci NFC telefonu. Po aktivaci je vyzván k přiložení telefonu k libovolnému zařízení s NFC čipem. Po detekci čipu je test ukončen. Test může ukončit i časomíra nastavená na třicet sekund.

2.3.2.20 GPS Test Ověří funkčnost GPS zařízení.

Aplikace vyzve uživatele k zapnutí GPS. Poté začne zjišťovat polohu a porovnávat ji s polohou nastavenou v nastavení a zároveň ukazuje na displeji mapu oblohy s umístěním připojených družic. Ve chvíli, kdy zjišťovaná poloha souhlasí s polohou v nastavení, test se ukončí. Test může být také ukončen časovačem nastaveným na čtyřicet pět sekund.

2.3.2.21 Přední mikrofon Pomocí tohoto testu je ověřen přední mikrofon. Přední mikrofon je ten, který slouží k telefonování.

Aplikace vyzve uživatele, nechť spustí tlačítkem nahrávání a začne mluvit do mikrofonu v přední spodní části telefonu. Během nahrávání se na displeji zobrazuje, jak dlouho se už nahrává. Po deseti sekundách nahrávání se nahrávání zastaví a nahraný zvuk se začne přehrávat z reproduktoru telefonu. Uživatel může přepnout na přehrávání zvuku ze sluchátka telefonu. Aplikace se zeptá uživatele, jestli jde zvuk zřetelně slyšet.

2.3.2.22 Zadní mikrofon Test zadního mikrofonu funguje obdobně jako test předního mikrofonu, viz 2.3.2.21, až na to, že je uživatel vyzván k nahrávání zvuku zadním mikrofonem, ten slouží k nahrávání zvuku při natáčení videa kamerou.

2.3.2.23 RAM Tento test zjistí zaplněnost operační paměti telefonu, někdy nesprávně označované zkratkou RAM.

Test probíhá automaticky bez zásahu uživatele. Změří zaplněnost operační paměti a skončí.

2.3.2.24 CPU Test měří zatížení procesoru v telefonu.

Test probíhá automaticky bez zásahů uživatele. Po dobu deseti sekund se měří zatížení procesoru. Z naměřených hodnot se poté vypočítá průměrná hodnota.

2.3.2.25 Zdířka na paměťovou kartu Test ověří funkčnost paměťové karty.

Aplikace vyzve uživatele k vložení paměťové karty do zařízení. Poté na paměťové kartě vytvoří textový soubor a přečte, co v něm je. Jestli zápis i čtení proběhlo úspěšně, vymaže soubor a test končí.

2.4 Rozbor hardwarových komponent telefonů a jejich softwarového vybavení

2.4.1 Samsung Galaxy S3 Neo

Tabulka 1: Obecné parametry

Parametr	Hodnota
Výrobce	Samsung
Označení	GT-I9301I
Operační systém	Android
Verze OS	4.4 (KitKat)
Hmotnost	132 g
Možnost paměťové karty	ano

Tabulka 2: Parametry displeje

Parametr	Hodnota
Rozlišení	1280 x 720
Velikost úhlopříčky	4.8"
Počet barev	16 miliónů
Počet displejů	1

Tabulka 3: Rozměry

Parametr	Hodnota
Výška	136,6 mm
Šířka	70,7 mm
Hloubka	8,6 mm

Tabulka 4: Fotoaparát

Parametr	Hodnota
Počet fotoaparátů	2
Rozlišení zadního fotoaparátu	8 Mpx
Rozlišení předního fotoaparátu	1,9 Mpx
Funkce fotoaparátu	Blesk, Automatické ostření, Záznam HD videa

Tabulka 5: Baterie

Parametr	Hodnota
Kapacita baterie	2100 mAh
Pohotovostní doba	330 hodin
Doba hovoru	840 minut

Tabulka 6: Procesor a paměť

Parametr	Hodnota
Frekvence procesoru	1,4 Ghz
Počet jader procesoru	4
Velikost úložiště	16 GB
Velikost operační paměti	1536 MB
Maximální velikost paměťové karty	64 GB

2.4.1.1 Senzory

1. Senzor zrychlení
2. Magnetický senzor
3. Gyroskop
4. Světelný senzor
5. Senzor přiblížení

2.4.2 Asus ZenFone 2 Laser

Tabulka 7: Obecné parametry

Parametr	Hodnota
Výrobce	Asus
Označení	ZE500KL
Operační systém	Android
Verze OS	5.0 (Lollipop)
Hmotnost	140 g
Možnost paměťové karty	ano

Tabulka 8: Parametry displeje

Parametr	Hodnota
Rozlišení	1280 x 720
Velikost úhlopříčky	5"
Počet barev	16 miliónů
Počet displejů	1

Tabulka 9: Rozměry

Parametr	Hodnota
Výška	143,7 mm
Šířka	71,5 mm
Hloubka	10,5 mm

Tabulka 10: Fotoaparát

Parametr	Hodnota
Počet fotoaparátů	2
Rozlišení zadního fotoaparátu	13 Mpx
Rozlišení předního fotoaparátu	5 Mpx
Funkce fotoaparátu	Blesk, Automatické ostření pomocí laseru, Záznam HD videa

Tabulka 11: Procesor, paměť a baterie

Parametr	Hodnota
Frekvence procesoru	1,2 Ghz
Počet jader procesoru	4
Velikost úložiště	16 GB
Velikost operační paměti	2048 MB
Maximální velikost paměťové karty	64 GB
Kapacita baterie	2070 mAh

2.4.2.1 Senzory

1. Senzor zrychlení
2. Magnetický senzor
3. Světelný senzor
4. Senzor přiblížení

3 Rozbor aplikace

Zpracováno v neveřejné části bakalářské práce.

4 Verifikace dosažených výsledků

Zpracováno v neveřejné části bakalářské práce.

5 Závěr

V první části práce jsem teoreticky rozebral problematiku týkající se operačního systému Android. Popsal jsem, z jakých vrstev se tento systém skládá a jak jednotlivé vrstvy fungují. Také jsem objasnil problematiku programovacích prostředí, ve kterých je možné programovat aplikace pro operační systém Android. V další části jsem popsal cíle, jakých chci při tvorbě aplikace dosáhnout. Protože cílem práce bylo vyvinout funkční aplikaci, popsal jsem jednotlivé části této aplikace a vazby mezi jejími částmi. Dále jsem rozebral hardwarové komponenty mobilů, pro které byla tato aplikace vyvinuta. Aplikace byla vyvinuta pro firmu Pegatron, a jelikož si tato firma nepřála, aby detaily aplikace byly zveřejněny, musí být část práce neveřejná. V neveřejné části práce jsou detailně popsány jednotlivé části aplikace, jejich funkčnost a způsob, jakým testují mobilní přístroj. Součástí neveřejné části jsou i výsledky testování této aplikace. Nicméně aplikace byla zatím testována jen na plně funkčních zařízeních. Důležitější bude nadcházející testování v reálném provozu na telefonech s vadnou součástí. Toto testování pravděpodobně odhalí nedostatky v aplikaci, které bude třeba opravit.

Literatura

- [1] Android (operační systém). In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2015 [cit. 2016-01-13]. Dostupné z: [https://cs.wikipedia.org/wiki/Android_\(operační_systém\)](https://cs.wikipedia.org/wiki/Android_(operační_systém))
- [2] SMIEH. Anatomy Physiology of an Android. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2012 [cit. 2016-01-14]. Dostupné z: <https://commons.wikimedia.org/wiki/File:Android-System-Architecture.svg>
- [3] Android Runtime. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2015 [cit. 2016-01-14]. Dostupné z: https://en.wikipedia.org/wiki/Android_Runtime
- [4] Android Runtime. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2015 [cit. 2016-01-14]. Dostupné z: https://cs.wikipedia.org/wiki/Android_Runtime
- [5] LINDER, Brad. What's new in Android 5.0 Lollipop? In: *Liliputing* [online]. 2014 [cit. 2016-01-14]. Dostupné z: <http://liliputing.com/2014/10/whats-new-android-5-0-lollipop.html>
- [6] Android Studio. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2015 [cit. 2016-01-14]. Dostupné z: https://cs.wikipedia.org/wiki/Android_Studio
- [7] Android software development. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2016 [cit. 2016-01-14]. Dostupné z: https://en.wikipedia.org/wiki/Android_software_development
- [8] Eclipse (software). In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2015 [cit. 2016-01-14]. Dostupné z: [https://en.wikipedia.org/wiki/Eclipse_\(software\)](https://en.wikipedia.org/wiki/Eclipse_(software))
- [9] Eclipse (vývojové prostředí). In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2015 [cit. 2016-01-14]. Dostupné z: [https://cs.wikipedia.org/wiki/Eclipse_\(vývojové_prostředí\)](https://cs.wikipedia.org/wiki/Eclipse_(vývojové_prostředí))
- [10] Desktop IDEs. *Eclipse* [online]. Ottawa (Kanada): Eclipse Foundation, 2016 [cit. 2016-01-14]. Dostupné z: <http://www.eclipse.org/ide/>
- [11] VÁVRŮ, Jiří a Miroslav UJBÁNYAI. *Programujeme pro Android. 2., rozš. vyd.* Praha: Grada, 2013, 250 s. Průvodce (Grada). ISBN 9788024748634.

- [12] SensorEvent: Class Overview. In: *Developers* [online]. 2016 [cit. 2016-04-22]. Dostupné z: <http://developer.android.com/reference/android/hardware/SensorEvent.html>
- [13] LEOW, Peter. Android Connectivity: Exploring the connectivity options in Android. In: *Code project: For those who code* [online]. Toronto Ontario: CodeProject, 2014 [cit. 2016-04-25]. Dostupné z: <http://www.codeproject.com/Articles/814814/Android-Connectivity>
- [14] NITSCH, Sascha. /proc/stat explained. In: *Www.LinuxHowtos.org: howtos, tips&tricks and tutorials for linux* [online]. Frankfurt nad Mohanem: Unternehmensberatung UG, 2013 [cit. 2016-04-25]. Dostupné z: <http://www.linuxhowtos.org/System/procstat.htm>